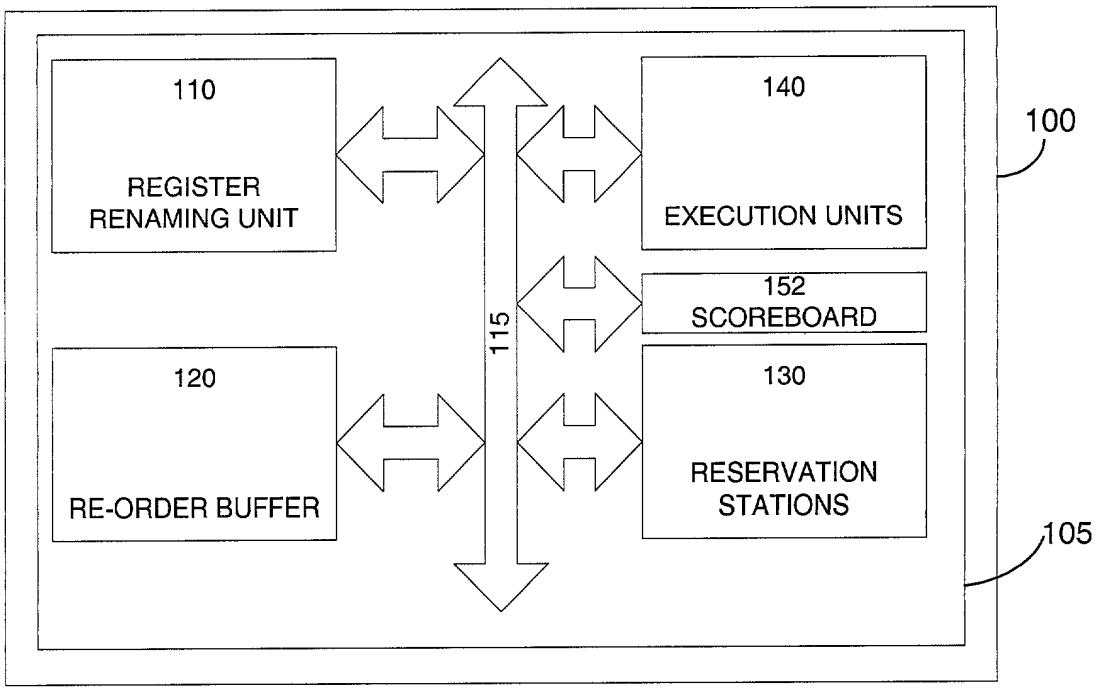
110
REGISTER
RENAMING UNIT
140
EXECUTION UNITS
100
152
SCOREBOARD
115
120
RE-ORDER BUFFER
130
RESERVATION
STATIONS
105

Fig. 1

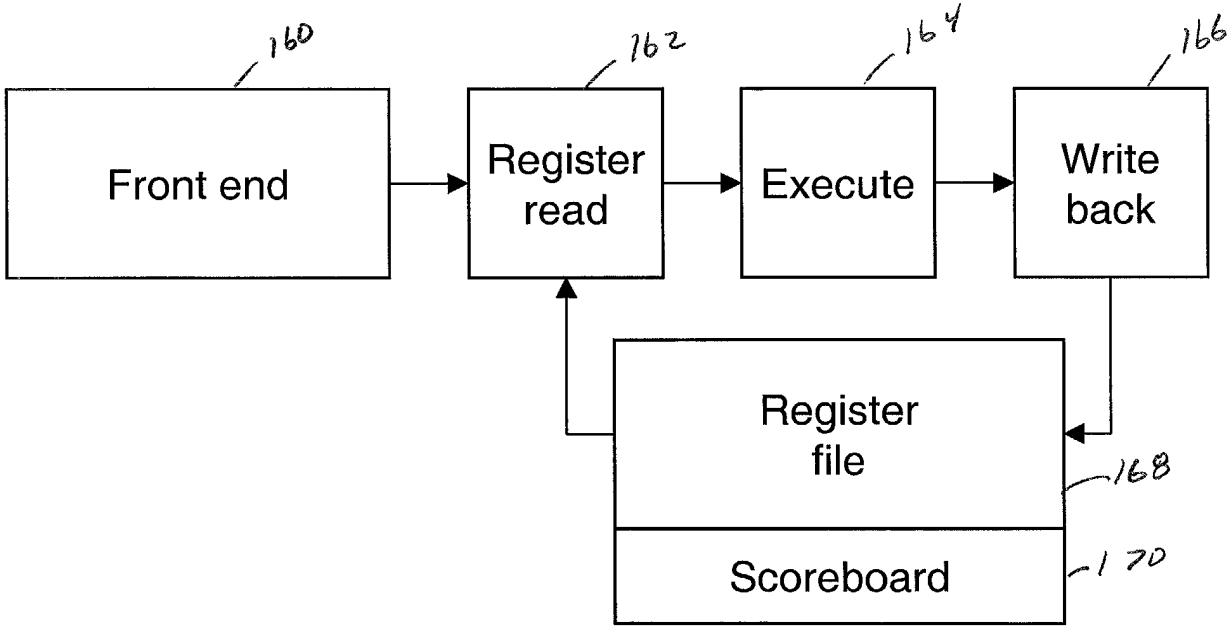
160
162
164
166
Front end
Register read
Execute
Write back
Register file
168
Scoreboard
170

Fig. 1A

000E2F "T98T5260

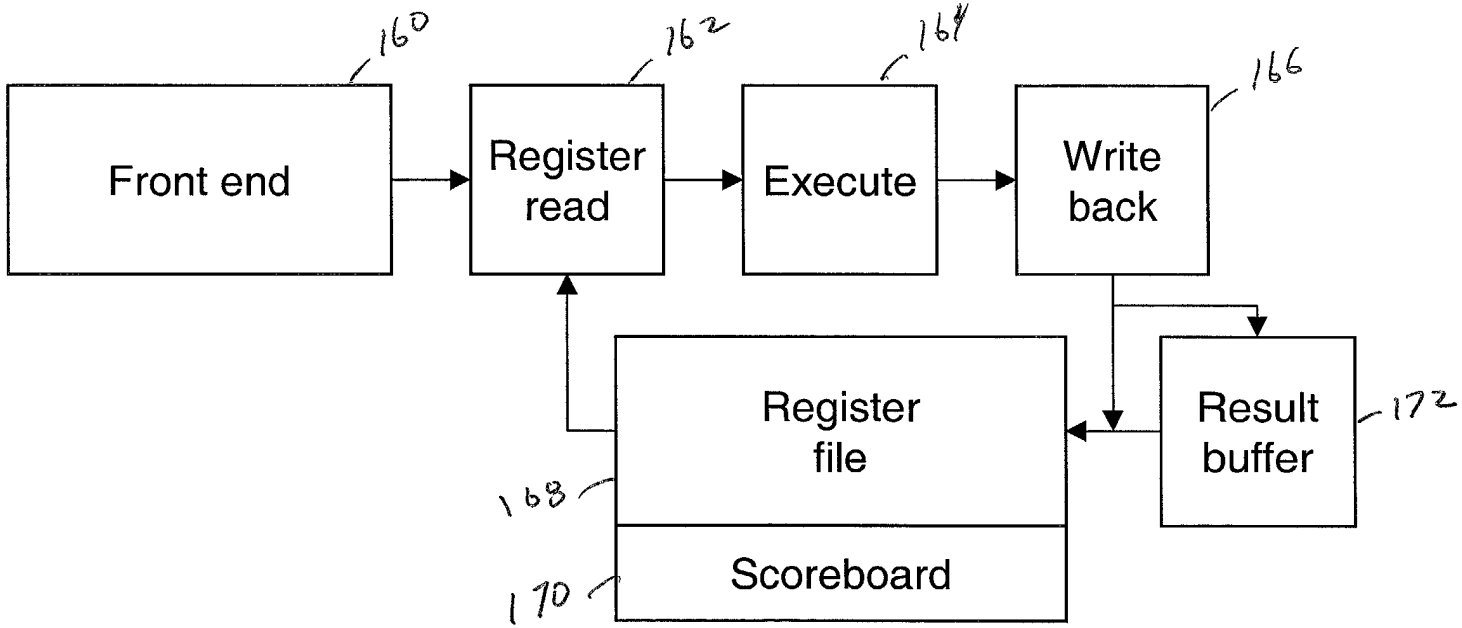

Fig. 1B

Docket: T9875260

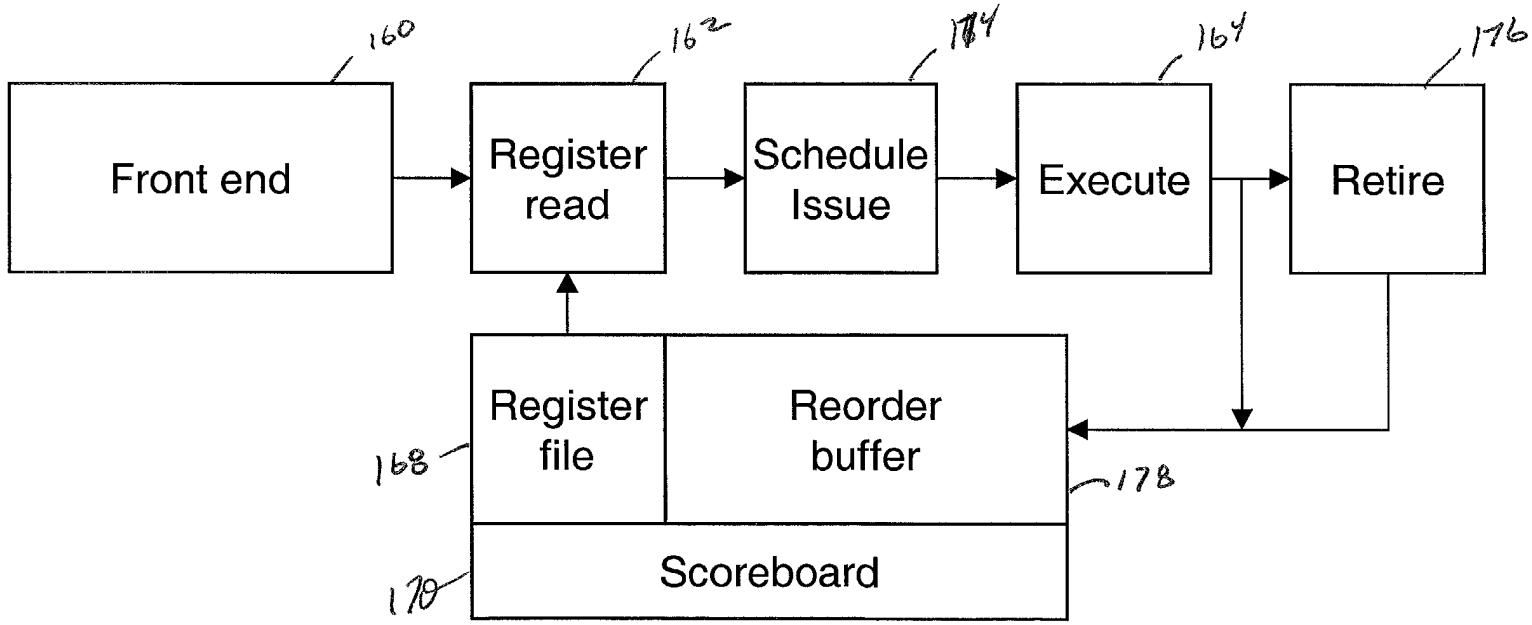

Fig. 1C

INST#		INSTRUCTION	DESCRIPTION
1:		mov 0 -> R1	# establishes R1
2:		ld (mem) -> R2	# loads R2 from memory
3:		cmp 0, R2 -> P1	# compares R2 to 0 and sets resulting predicate P1
4:	(P1)	add 5, R1 -> R1	# adds 5 to R1 if P1 is true
5:		sub 10, R1 -> R3	# sets R3 with the difference of 10 and R1

Fig. 2

Docket No. 42390.P13659

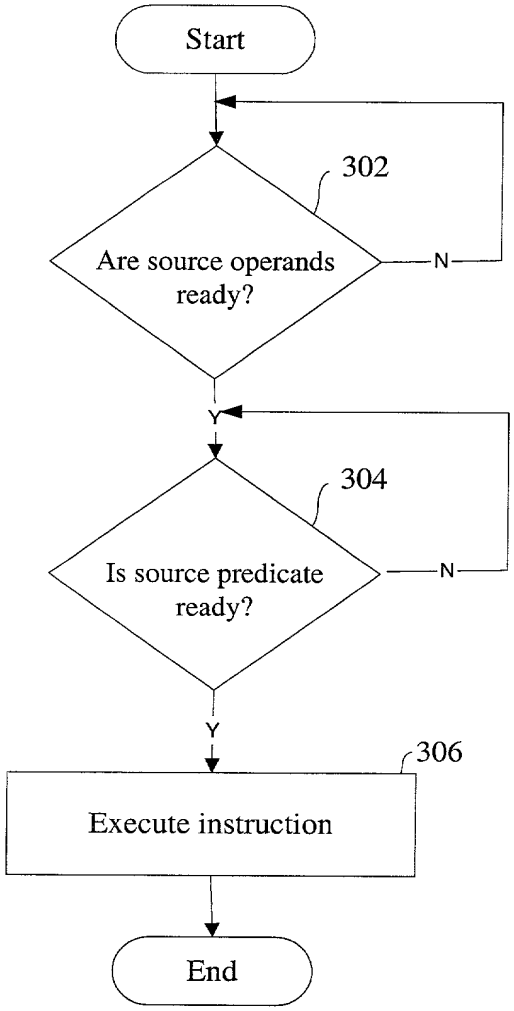
Start
302
Are source operands ready?
N
Y
304
Is source predicate ready?
N
Y
306
Execute instruction
End

FIG. 3A

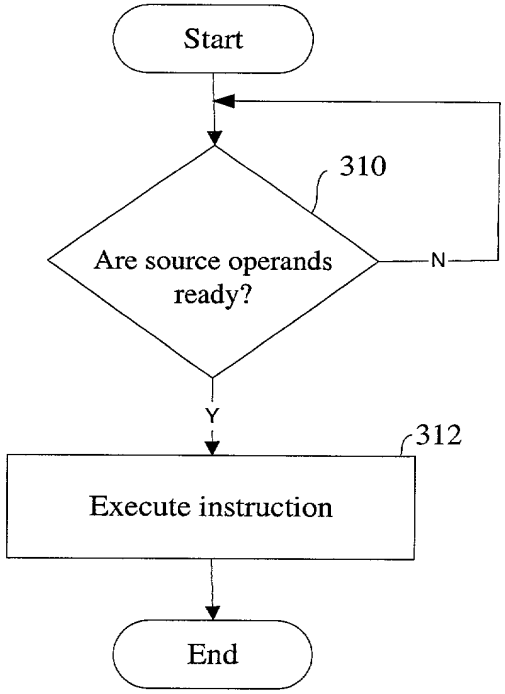
Start
310
Are source operands ready?
N
Y
312
Execute instruction
End

FIG. 3B

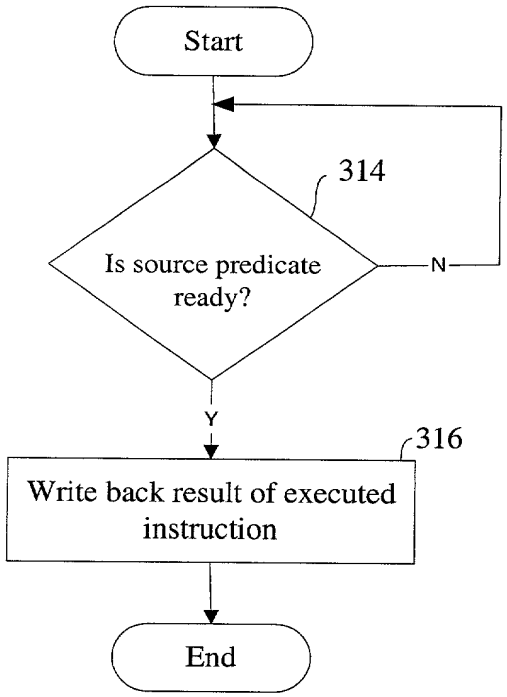
Start
314
Is source predicate ready?
N
Y
316
Write back result of executed instruction
End

FIG. 3C

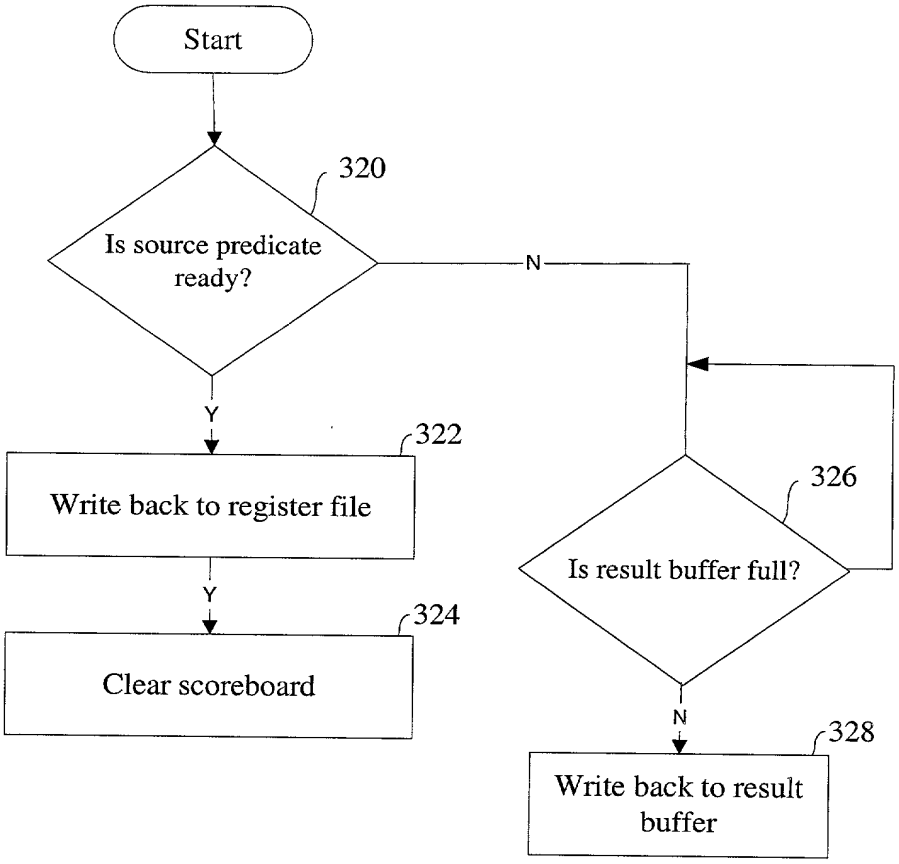
Start
320
Is source predicate ready?
N
Y
322
Write back to register file
Y
324
Clear scoreboard
326
Is result buffer full?
N
328
Write back to result buffer

FIG. 3D

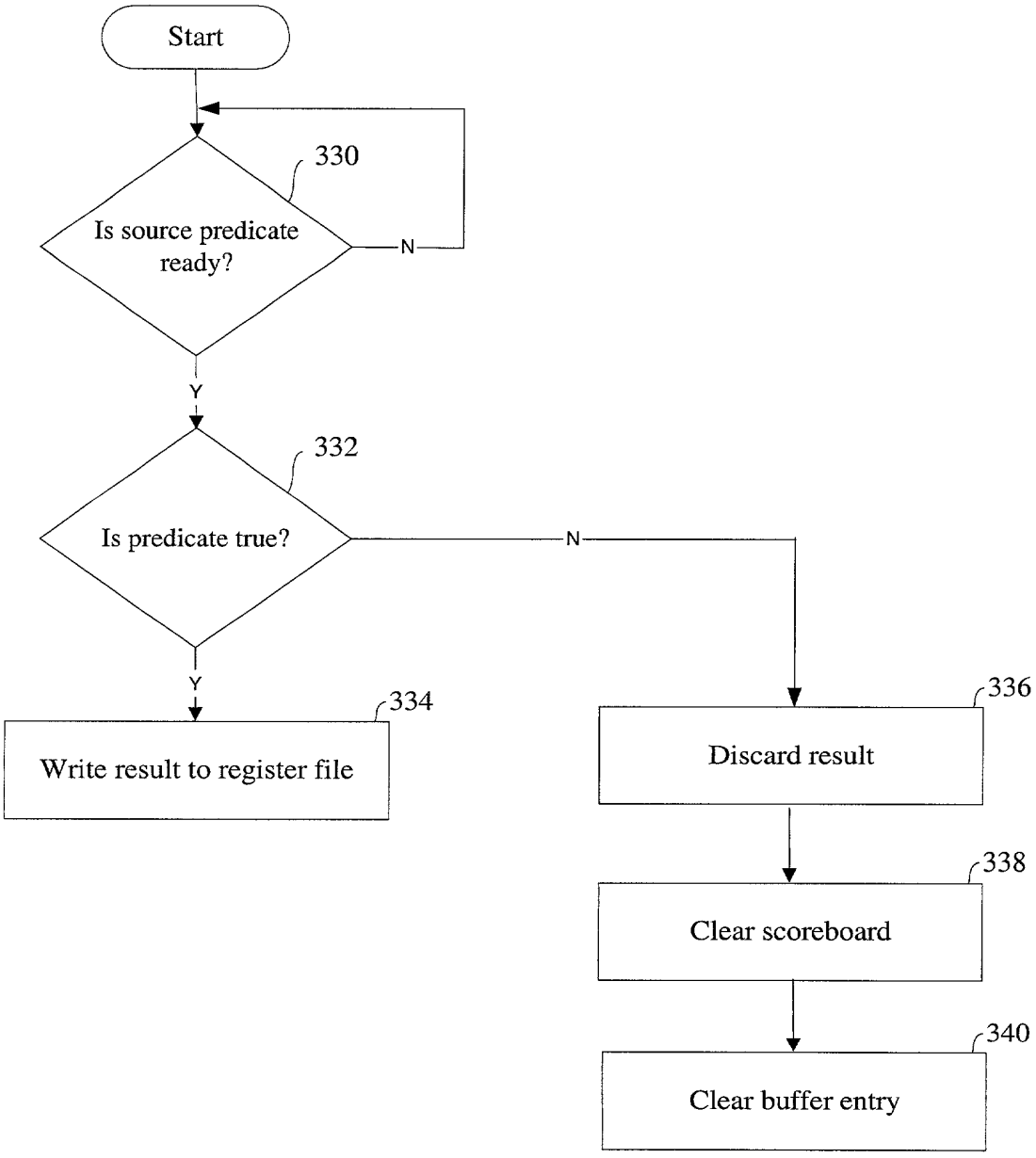
Start
330
Is source predicate ready?
N
Y
332
Is predicate true?
N
Y
334
Write result to register file
336
Discard result
338
Clear scoreboard
340
Clear buffer entry

FIG. 3E

Docket No. 42390.P18908

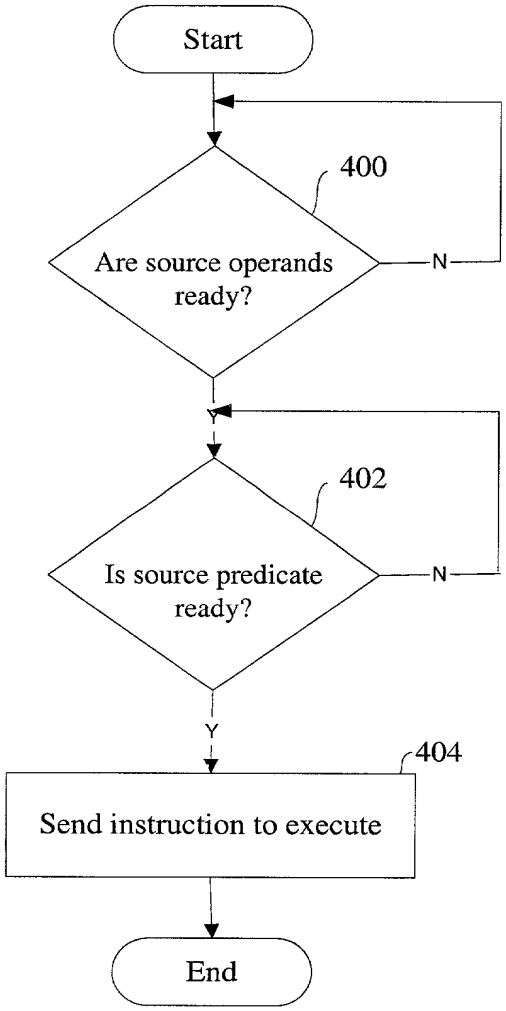

FIG. 4A

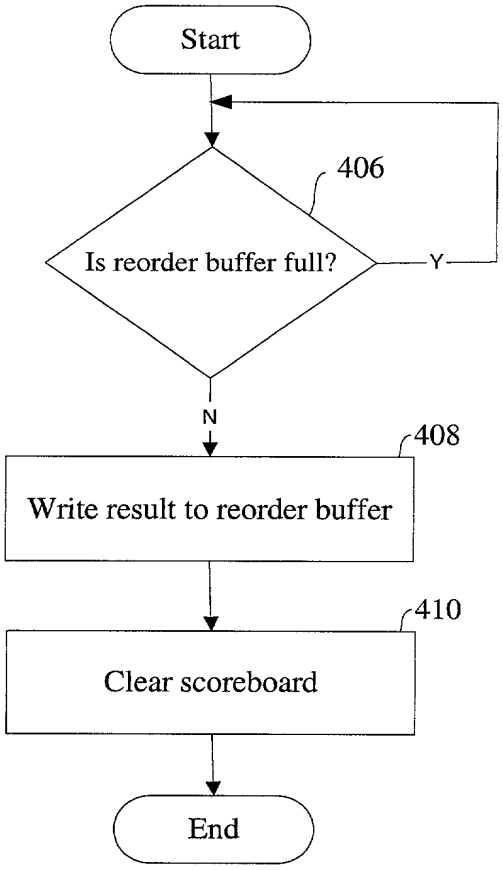

FIG. 4B

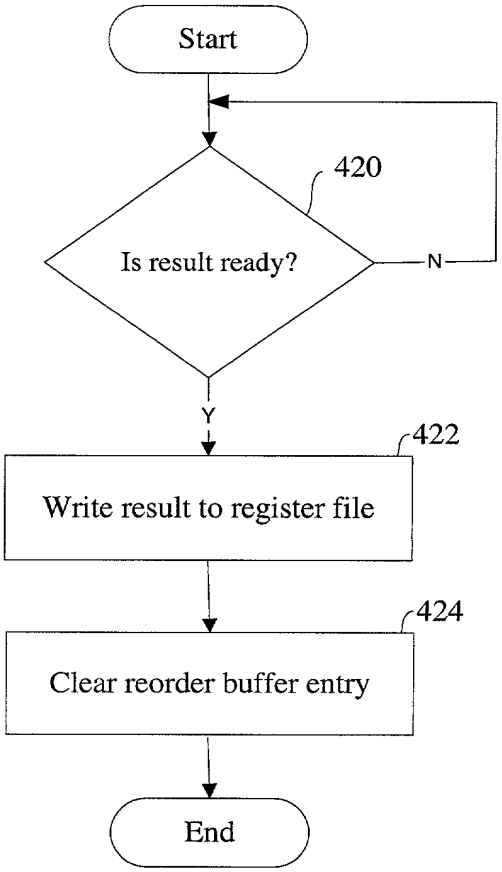
Start
420
Is result ready?
N
Y
422
Write result to register file
424
Clear reorder buffer entry
End

FIG. 4C

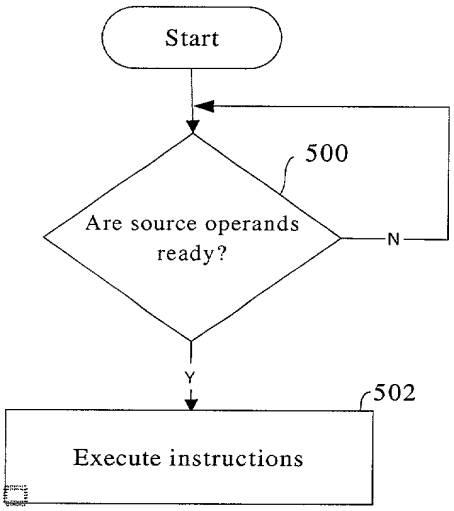
Start
500
Are source operands ready?
N
Y
502
Execute instructions

FIG. 5A

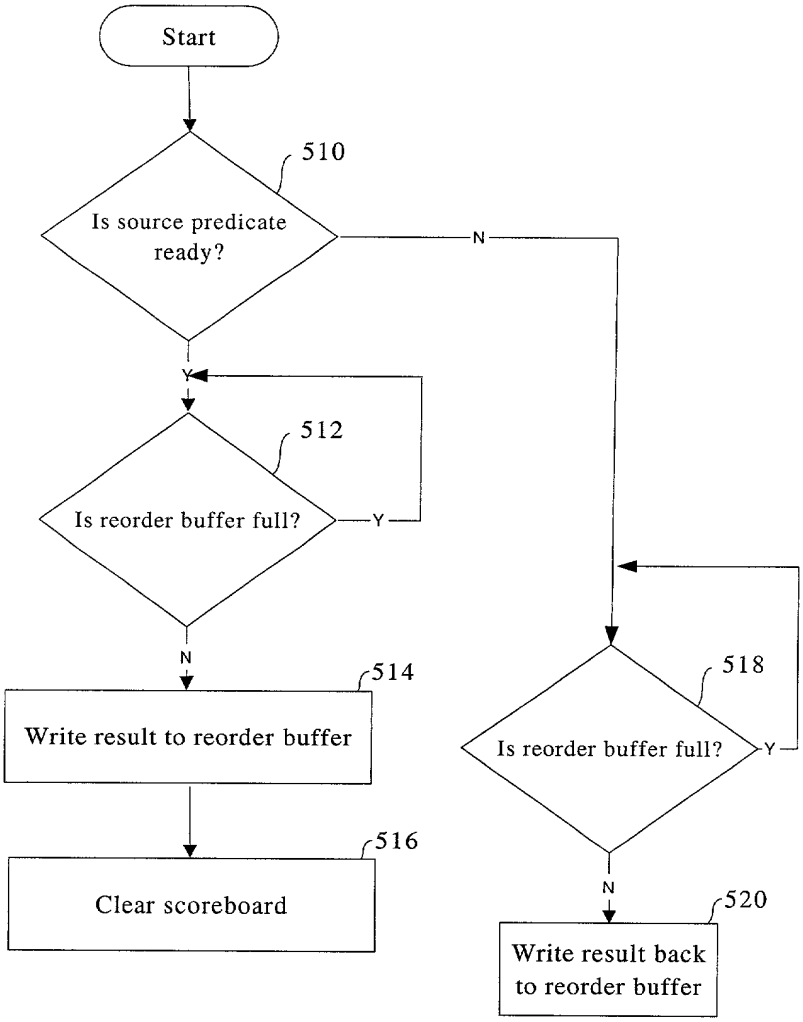
Start
510
Is source predicate ready?
N
Y
512
Is reorder buffer full?
Y
N
514
Write result to reorder buffer
516
Clear scoreboard
518
Is reorder buffer full?
Y
N
520
Write result back to reorder buffer

FIG. 5B

Docket: 1986.5760

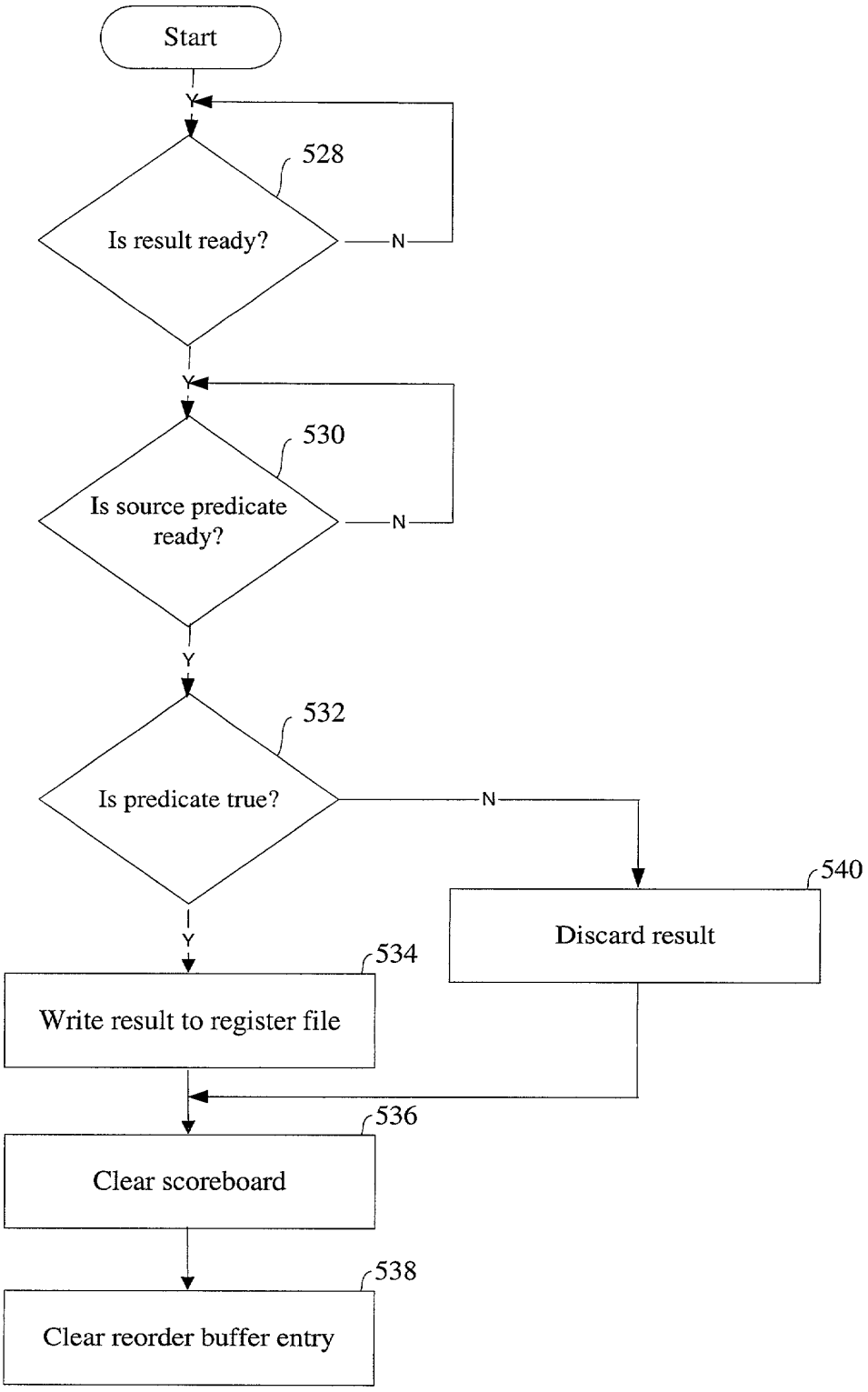

FIG. 5C